Electronic Version
Stylesheet Version v1.1.1

Description

CLOCK AND DATA RECOVERY CIRCUIT

BACKGROUND OF INVENTION

[0001] 1. Field of the Invention

[0002] The present invention relates to serial data communications, and more specifically, to a clock and data recovery circuit (CDR) used in a serial data communication system.

[0003] 2. Description of the Prior Art

[0004] Compared with parallel data communications, serial data communications are small in size and have a longer transmission distances. Although slower than parallel data communications, recently developed serial data communication devices such as USB1.1 and USB2.0 have solved those disadvantages, wherein the transmission speed of USB1.1 is up to 12Mbps, and USB2.0 up to 480Mbps.

[0005] Please refer to Fig.1 showing a conventional serial data communication system. The serial data communication system 10 includes a transmitter 12 for transmitting data, a serial bus 14 connected to the transmitter 12 for trans-

mitting data, and a receiver 16 for receiving data from the serial bus 14. Please refer to Fig.2 showing a waveform diagram concerning the output data $DATA_{out}$ transmitted by the transmitter 12 and input data $DATA_{in}$ received by the receiver 16 in the serial data communication system 10. As shown in Fig.2, the input data $DATA_{in}$ received by the receiver 16 is not synchronized with the output data $DATA_{out}$ transmitted by the transmitter 12. In other words, the phases of the input data $DATA_{in}$ and the output data $DATA_{out}$ are different. Therefore, the receiver 16 needs to include a CDR 20 to adjust the phase difference between the input data $DATA_{in}$ and the output data $DATA_{out}$, in order to correctly read the input data $DATA_{in}$.

[0006] The receiver 16 shown in Fig.1 includes a front amplifier 18 for signal amplifying, and the CDR 20. The CDR 20 can be a hybrid CDR combining the advantages of high data transmission speed analog CDR and low noise interference digital CDR. The CDR 20 generates corresponding recovery data $DATA_{rd}$ and a re-time clock CLK_{rt} according to the input data $DATA_{in}$. The CDR 20 includes a phase shifter 22 for generating a plurality of discrete clocks CLK_{dis} at different phases according to a reference clock CLK_{ref} (e.g. the phase shifter 22 generates 24 discrete clocks

CLK_0-CLK_{345} at different phases. In other words, any two consecutive discrete clocks CLK_{dis} have a phase difference of 15 degrees.). A counter 24 is used for counting the number of rising edges from "0" to "1" to determine whether to start sampling, a data sampler 26 for receiving the 24 discrete clocks CLK_0-CLK_{345} and the input data $DATA_{in}$ and outputting a select signal CS accordingly (the select signal CS indicates the period when between two consecutive discrete clocks CLK_{dis} among the 24 discrete clocks CLK_0-CLK_{345} the rising edge of the input data $DATA_{in}$ occurs). A phase selector 28 is electrically connected to the data sampler 26, a multiplexer 30 is used for selecting one from the 24 discrete clocks CLK_0-CLK_{345} according to a phase select signal PS output by the phase selector 28, and a phase detector 32 is for modifying the phase select signal PS output by the phase selector 28 according to the phase difference between a selected clock CLK_{cs} output by the multiplexer 30 and the input data $DATA_{in}$. The frequency of the reference clock CLK_{ref} of the phase shifter 22 is approximately the same as that of the output data $DATA_{out}$ transmitted by the transmitter 12.

[0007] Please refer to Fig.3 showing a circuit diagram of the data sampler 26 in the CDR 20. The data sampler 26 includes

24 D flip-flops 34 with all their clock input ends CLK electrically connected to the input data $DATA_{in}$, and their signal input ends D respectively electrically connected to the discrete clocks CLK_0-CLK_{345} generated by the phase shifter 22. A signal output end Q of the D flip-flop 34 shows that the rising edge of the input data $DATA_{in}$ occurs between two consecutive discrete clocks among the 24 discrete clocks CLK_0-CLK_{345}. For instance, if the rising edge of the input data $DATA_{in}$ occurs between discrete clocks CLK_{135} and CLK_{150}, the select signal CS output by the data sampler 26 for this example is 003FFFx. This means the discrete clock CLK_{dis} selected by the multiplexer 30 is CLK_{150} (or CLK_{135}).

[0008] The operation of the CDR 20 occurs after the number of the rising edges of the input data $DATA_{in}$ counted by the counter 24 exceeds a predetermined value, i.e. under a stable condition and is described as follows. After detecting that the rising edge of the input data $DATA_{in}$ occurs between the discrete clocks CLK_{135} and CLK_{150}, the data sampler 26 generates the select signal CS (003FFFx) corresponding to the discrete clock $CLK_{150.}$ The phase selector 28 then generates the phase select signal PS according to the select signal CS and a calibration signal CR gener-

ated by the phase detector 32. This controls the multiplexer 30 to output one from the discrete clocks CLK_{135}, CLK_{150}, CLK_{165} to be the selected clock CLK_{cs}. Eventually, the selected clock CLK_{cs} output from the multiplexer 30 becomes the re-time clock CLK_{rt}, the result being that the re-time clock CLK_{rt} triggers the input data $DATA_{in}$ to form the recovery data $DATA_{rd}$.

[0009] During the transmission of the discrete clock CLK_{dis} generated by the phase shifter 22 toward the multiplexer 30, phase deviation is inevitable. Therefore, the selected clock CLK_{cs} output by the multiplexer 30 differs from an ideal discrete signal CLK_{ideal}corresponding to the input data $DATA_{in}$. Thus the selected clock CLK_{cs} output by the multiplexer 30 is not necessarily allow the recovery data $DATA_{rd}$ corresponding to the input data $DATA_{in}$. The phase detector 32 is for further modifying the select signal CS output by the data sampler 26 according to the phase relationship between the selected clock CLK_{cs} and the input data $DATA_{in}$. This allows the phase selector 28 to generate the phase select signal PS in order to further control the multiplexer 30 to output the selected clock CLK_{cs} or a discrete clock CLK_{dis} previous or next to the selected clock CLK_{cs}. More clearly, if the phase detector 32

detects that the selected clock CLK_{cs} lags the input data $DATA_{in}$, the calibration signal CR generated by the phase detector 32 is accumulated on a following select signal CS generated by the data sampler 26 in order to form the phase select signal PS. For instance, if the discrete clock CLK_{180} output by the multiplexer 30 (i.e. the selected clock CLK_{cs}) lags behind the input data $DATA_{in}$, the calibration signal CR generated by the phase detector 32 is accumulated on the select signal CS generated by the data sampler 26 in order to form the phase select signal PS. That is, the multiplexer 30 should originally output the discrete clock CLK_{180} under control of a select signal CS generated by the data sampler 26 according to following input data $DATA_{in}$, however, due to the accumulation of the calibration signal CR, the multiplexer 30 outputs a discrete clock CLK_{195} instead. Conversely, if the discrete clock CLK_{180} output by the multiplexer 30 (i.e. the selected clock CLK_{cs}) leads the input data $DATA_{in}$, the calibration signal CR generated by the phase detector 32 is decreased from the select signal CS generated by the data sampler 26 in order to form the phase select signal PS. That is, the multiplexer 30 should originally output the discrete clock CLK_{180} under control of a select signal CS

generated by the data sampler 26 according to following input data $DATA_{in}$, however, due to the accumulation of the calibration signal CR, the multiplexer 30 outputs a discrete clock CLK_{165} instead.

[0010] As for the CDR 20, the number of the discrete clocks CLK_{dis} generated by the phase shifter 22 directly relates to a phase jitter tolerable by the input data $DATA_{in}$. That is, the more discrete clocks CLK_{dis} generated by the phase shifter 22, the more synchronized the re-time clock CLK_{rt} generated by the CDR 20 to the input data $DATA_{in}$. Accordingly, the more phase jitter the input data $DATA_{in}$ can tolerate, the more accurate the recovery data $DATA_{rd}$ is, and accordingly the lower bit error rate (BER) the recovery data $DATA_{rd}$. However, in order to read the input data $DATA_{in}$ as accurately as possible, the data sampler 26 in the CDR 20 is required to include enough D flip-flops (or any implementation having phase delay circuits). These D flip-flops not only occupy too much area, but also consume too much power.

SUMMARY OF INVENTION

[0011] It is therefore a primary objective of the present invention to provide a CDR to reduce the number of D flip-flops in a data sampler, in order to solve the problems mentioned

above.

[0012] Briefly summarized, a clock and data recovery circuit (CDR) generates a recovery clock according to input data and a reference clock corresponding to the input data. The CDR includes a phase shifter generating M discrete clocks at different phases according to the reference clock; a data sampler generating a select signal according to the input data and the M discrete clocks; a primary phase selector outputting two consecutive discrete clocks and at least one interpolated clock with a phase between the phases of the two consecutive discrete clocks according to the select signal, a multiplexer selecting one from the two consecutive discrete clocks and the interpolated clock to be a selected output clock, a phase detector receiving the selected output clock to be the recovery clock, and outputting an advanced calibration signal if the recovery clock leads or lags behind the input data, an advanced phase selector receiving the advanced calibration signal and transmitting the phase select signal to the multiplexer for adjusting the selection of the selected clock, and a primary calibration signal to the primary phase selector for adjusting the two consecutive discrete clocks and at least one interpolated clock corresponding

to them.

[0013] These and other objectives of the present invention will no doubt become obvious to those of ordinary skill in the art after reading the following detailed description of the preferred embodiment that is illustrated in the various figures and drawings.

BRIEF DESCRIPTION OF DRAWINGS

[0014] Fig.1 illustrates a conventional serial data communication system.

[0015] Fig.2 is a waveform diagram concerning the output data transmitted by the transmitter and input data received by the receiver in the conventional serial data communication system.

[0016] Fig.3 is a circuit diagram of the data sampler in the conventional CDR.

[0017] Fig.4 is a block diagram of a CDR according to the present invention.

[0018] Fig.5 is a circuit diagram of the data sampler in the CDR according to the present invention.

[0019] Fig.6 illustrates the variation of the phase select signal according to the present invention.

[0020] Fig.7 illustrates a circuit diagram of the primary phase selector in the CDR.

DETAILED DESCRIPTION

[0021] A phase shifter in a CDR according to the present invention generates M discrete clocks CLK_{dis} wherein M is less than the number required by the prior art. At least one interpolated clock CLK_{int} is then found from any two consecutive discrete clocks CLK_{dis} by interpolation, and a set of clocks is formed by it and the two consecutive discrete clocks CLK_{dis}. Subsequently, one clock CLK_{cs} is selected being more synchronized to input data $DATA_{in}$ from the set of clocks. Since finding at least one interpolated clock CLK_{int} by interpolation requires only one common circuit, a large number of D flip-flops as in the prior art are no longer required to implement the data sampler. So that the number of the D flip-flops, and accordingly the area occupied by the D flip-flops and the cost are effectively reduced.

[0022] Please refer to Fig.4 showing a block diagram of a CDR 50 according to the present invention. The CDR 50 includes a phase shifter 52, a data sampler 56 electrically connected to the phase shifter 52, a primary phase selector 58 electrically connected between the phase shifter 52 and the data sampler 56, a multiplexer 60 electrically connected to the primary phase selector 58, a phase detector 62

electrically connected to the multiplexer 60, a counter 54 electrically connected between the data sampler 56 and the phase detector 62, and an advanced phase selector 64 electrically connected to the multiplexer 60, the primary phase selector 58 and the phase detector 62.

[0023] The phase shifter 52 is an analog phase-locked loop (APLL) or a delay-locked loop (DLL), which generates a plurality of discrete clocks CLK_{dis} at different phases according to a reference clock CLK_{ref}. In the present invention, since the discrete clocks are generated by interpolation, the phase shifter 52 is only required to generate 8 discrete clocks CLK_0 to CLK_{315} at different phases. That is, any two consecutive discrete clocks CLK_{dis} have a phase difference of 45 degrees. The data sampler 56 generates a select signal CS according to where the rising edges of input data $DATA_{in}$ occur. Please refer to Fig.5 showing a circuit diagram of the data sampler 56 in the CDR 50 according to the present invention. The data sampler 56 is structurally similar to the data sampler 26 in the conventional CDR 20, however, the data sampler 56 samples the 8 discrete clocks CLK_0 to CLK_{315} using the input data $DATA_{in}$ to output the select signal CS. Because the multiplexer 60, the phase detector 62 and the counter 54 have

the same functions as the multiplexer 30, the phase detector 32 and the counter 24 in the conventional CDR 20, a further description is hereby omitted.

[0024] The operation of the CDR 50 after the number of the rising edges of the input data $DATA_{in}$ counted by the counter 54 exceeds a predetermined value is described as follows. (e.g. After a second and a third data, predetermined values of a primary calibration signal CR_p and a phase select signal PS output from the advanced phase selector 64 have been set up as described in the following.) After detecting that the rising edge of the input data $DATA_{in}$ occurs between discrete clocks CLK_{135} and CLK_{180}, the data sampler 56 generates the select signal CS corresponding to the discrete clock CLK_{180} (or CLK_{135}). The primary phase selector 58 then outputs the discrete clocks CLK_{135} and CLK_{180} (two consecutive discrete clocks CLK_{dis}) and discrete clocks CLK_{150} and CLK_{165} interpolated from the discrete clocks CLK_{135} and $CLK_{180.}$ (The existence of the discrete clocks CLK_{150} and CLK_{165} means that there is at least one interpolated clock CLK_{int} interpolated from the two consecutive discrete clocks $CLK_{dis.}$) The clocks are output according to the select signal CS and the primary calibration signal CR_p generated by the advanced phase

selector 64. The multiplexer 60 selects a selected clock CLK_{cs} from the discrete clock CLK_{135}, the interpolated clock CLK_{150}, the interpolated clock CLK_{165}, or the discrete clock CLK_{180}. And eventually, the selected clock CLK_{cs} output from the multiplexer 60 becomes a real-time clock CLK_{rt}. The result of the real-time clock CLK_{rt} triggering the input data $DATA_{in}$ is recovery data $DATA_{rd}$.

[0025] Similarly, the phase detector 62 in the CDR 50 outputs signals relating to modifying the selected clock CLK_{cs} of the multiplexer 60 according to the phase difference between the selected clock CLK_{cs} and the input data $DATA_{in}$. In the present invention, the modified signal output by the phase detector 62 is an advanced calibration signal CR_a.

[0026] Please refer to Fig.6 showing the variation of the phase select signal PS according to the present invention. Assume the predetermined value of the phase select signal PS is 10b. That is, the multiplexer 60 outputs a second leading discrete clock CLK_{dis} (CLK_{165} in this case) from the four discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180} according to the phase select signal PS (10b). If the phase detector 62 detects that the discrete clock CLK_{165} (i.e. the selected clock CLK_{cs}) lags behind the input data $DATA_{in}$, the phase detector 62 outputs the advanced calibration

signal CR_a to increment the phase select signal PS by one (the phase select signal PS is modified into 11b). In this way, the multiplexer 60 outputs the most leading discrete clock CLK_{dis} (CLK_{180} in this case) from the four discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180}. Assume the predetermined value of the phase select signal PS is 10b and the phase detector 62 detects that the discrete clock CLK_{165} (i.e. the selected clock CLK_{cs}) leads the input data $DATA_{in}$. The phase detector 62 outputs the advanced calibration signal CR_a to decrement the phase select signal PS by one (the phase select signal PS is modified into 01b). In this way, the multiplexer 60 outputs a third leading discrete clock CLK_{dis} (CLK_{150} in this case) from the four discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180} instead.

[0027] If the phase select signal PS is already 11b (i.e. the value will overflow to 00b if 1 is added), and the phase detector 62 detects that the discrete clock CLK_{180} (i.e. the selected clock CLK_{cs}) lags behind the input data $DATA_{in}$, since there is no discrete clock CLK_{dis} leading the discrete clock CLK_{180} among the four discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180}, the advanced phase selector 64 outputs the primary calibration signal CR_p whenever the phase select signal PS overflows from 11b to 00b. The primary

phase selector 58 outputs discrete clocks CLK_{180}, CLK_{195}, CLK_{210}, CLK_{225} to the multiplexer 60 instead of the discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180}. At this time, because the multiplexer 60 is required to output the discrete clocks CLK_{195} (leading the discrete clock CLK_{180}), the phase select signal PS should be set to 01b, instead of rolling over from 11b to 00b. In other words, whenever the phase select signal PS overflows, the advanced phase selector 64 sets the phase select signal PS to 01b.

[0028] Conversely, if the phase select signal PS is 00b (i.e. the value will underflow if 1 is subtracted), and the phase detector 62 detects that the discrete clock CLK_{135} (i.e. the selected clock CLK_{cs}) leads the input data $DATA_{in}$, since there is no discrete clock CLK_{dis} lagging behind the discrete clock CLK_{135} among the four discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180}, the advanced phase selector 64 outputs the primary calibration signal CR_p whenever the phase select signal PS underflows from 00b to 11b. The primary phase selector 58 outputs discrete clocks CLK_{90}, CLK_{105}, CLK_{120}, CLK_{135} to the multiplexer 60 instead of the discrete clocks CLK_{135}, CLK_{150}, CLK_{165}, CLK_{180}. Since at this time, the multiplexer 60 is required to output the discrete clocks CLK_{120} (lagging behind the discrete clock

CLK_{135}), the phase select signal PS should be set as 10b, instead of rolling over from 00b to 11b. In other words, whenever the phase select signal PS underflows, the advanced phase selector 64 sets the phase select signal PS as 10b. Of course, the overflow, underflow and reset operations described above can be also implemented in other manners.

[0029] Please refer to Fig.7 showing a circuit diagram of the primary phase selector 58 in the CDR. Two different discrete clocks CLK_{dis1}, CLK_{dis2} form the primary phase selector through a combination of a plurality of inverters. The width/length (W/L) of inverters A and B inside the combination can be properly controlled to obtain the required interpolated clock signal CLK_{int}. For example, the W/L can be expanded to obtain more interpolated clock signals CLK_{int}. Since many interpolated clock signals CLK_{int} can be produced, D flip-flops (or devices) for generating discrete clocks in the phase shifter 52 and for sampling discrete clocks in the data sampler 56 can be effectively reduced.

[0030] In contrast to the prior art, the CDR 50 according to the present invention includes the phase shifter 52 generating only 8 discrete clocks CLK_0-CLK_{315}, and the data sampler 56 including only 8 D flip-flops, thus the CDR 50 is

smaller in size and consumes less power. Moreover, the primary phase selector 58 of the CDR 50 generates the plurality of interpolated clocks CLK_{int} based on the two consecutive discrete clocks CLK_{dis} generated by the phase shifter 52 as required, thus there is more elasticity on the CDR 50.

[0031] Those skilled in the art will readily observe that numerous modifications and alterations of the device may be made while retaining the teachings of the invention. Accordingly, the above disclosure should be construed as limited only by the metes and bounds of the appended claims.